

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Redakční systém se SEO propagací
Redaction System with SEO Propagation

2016

Jakub Bocek

Zadání bakalářské práce

Student:

Jakub Bocek

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Redakční systém se SEO propagací
Redaction System with SEO Propagation

Jazyk vypracování:

čeština

Zásady pro vypracování:

Vytvořte redakční systém obsahující tyto funkce: editace a vytváření obsahu stránek, hlavní menu, galerie multimediálních souborů (obrázky, video, zvuk) a kontaktní formulář. Kromě těchto funkcí bude systém obsahovat také napojení na centrální systém reklamy a SEO optimalizace. Tato funkce umožní zobrazení reklamy a odkazů na stránkách spravovaných vytvořeným redakčním systémem. Ke komunikaci mezi redakčním a reklamním systémem využijete REST.

1. Seznamte se s webovými technologiemi.
2. Podrobně popište techniky SEO optimalizace.
3. Proveďte rešerši podobných řešení.
4. Implementujte redakční systém, systém reklamy a SEO propagace.
5. Porovnejte implementovaný systém s dostupnými řešeními.

Seznam doporučené odborné literatury:

- [1] ENGE, Eric, et al. The art of SEO. O'Reilly Media, Inc., 2012.
- [2] ROSEN, Michael, et al. Applied SOA: service-oriented architecture and design strategies. John Wiley & Sons, 2008.
- [3] HOGAN, Brian P., et al. Web Development Recipes. Pragmatic Bookshelf, 2012.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jan Plucar**

Datum zadání: 01.09.2014

Datum odevzdání: 29.04.2016



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne: 29. 4. 2016



Podpis

Poděkování

Rád bych poděkoval vedoucímu bakalářské práce **Ing. Jan Plucar** za odbornou pomoc a konzultaci při vytváření této práce.

Abstrakt

Účelem této bakalářské práce je vytvořit jednoduchý redakční systém s podporou tvorby blogu uživatele. Na stránkách blogů se bude zobrazovat reklama zadaná v reklamním systému.

Redakční systém bude napojen na centrální systém reklamy, který umožní zobrazení reklamy na stránkách (blogy) vytvořených redakčním systémem.

Systém bude obsahovat SEO propagaci, která bude zajišťovat přední výsledky ve všech známých vyhledávačích, jakou jsou například Google nebo Seznam.

Klíčová slova

redakční systém, SEO propagace, systém reklamy

Abstract

The purpose of this bachelor thesis is to develop easy content management system with support of creating blog for user. On blogs will show advert specified in advertising system.

Content management system will be connect to a central system of advertising which enables to display adverts to pages (blogs) created by content management.

Content management system will contain SEO propagation which will provide a leading results in all known search engines for example Google and Seznam.

Keywords

content management system, SEO propagation, advertising system

Seznam použitých symbolů a zkratek

SEO - search engine optimization (optimalizace pro vyhledávače)

PPC - pay per click (platba za prokliknutí)

PPV - pay per view (platba za počet zobrazení)

RUP - rational unified proces (metodika vývoje softwaru)

UML - unified modeling language (grafický jazyk pro vizualizaci)

FURPS (model pro klasifikaci atributů kvality softwaru)

HTML – hypertext markup language (značkovací jazyk)

CSS – cascading style sheets (kaskádové styly)

Seznam obrázků

Obrázek č. 1 - Graf nejdražších klíčových slov

Obrázek č. 2 – Průměrné ceny klíčových slov

Obrázek č. 3 – PPC systém Google AdWords

Obrázek č. 4 – PPC systém Google AdWords zadání kampaně

Obrázek č. 5 – PPC systém Google AdWords pokračování zadání kampaně

Obrázek č. 6 – PPC systém Sklik

Obrázek č. 7 – PPC systém Sklik vytvoření kampaně

Obrázek č. 8 – PPC systém Sklik pokračování vytvoření kampaně

Obrázek č. 9 – PPC systém eTarget vytvoření odkazu

Obrázek č. 10 – Fáze projektu

Obrázek č. 11 – Use-case diagram

Obrázek č. 12 – Seznam reklam zákazníka

Obrázek č. 13 – Objednávka reklamy

Obrázek č. 14 – Třívrstvá architektura

Obrázek č. 15 – Diagram tříd

Obrázek č. 16 – Sekvenční diagram přihlášení

Obrázek č. 17 – Obecná abstraktní továrna

Obrázek č. 18 – Konkrétní abstraktní továrna

Obrázek č. 19 – Data mapper

Obrázek č. 20 – Technologie ASP. NET

Obrázek č. 21 – Ukázka běžící reklamy na blogu

Obrázek č. 22 – Formulář pro přidání reklamy do systému

Obsah

Úvod.....	- 1 -
1 Rozbor stávajících systémů.....	- 2 -
2 Postup vypracování.....	- 8 -
3 Analýza a návrh.....	- 10 -
3.1 Správa požadavků na systém podle metody FURPS.....	- 10 -
3.1.1 F – funkční požadavky.....	- 11 -
3.1.2 U – požadavky vhodnosti k použití.....	- 12 -
3.1.3 R – spolehlivostní požadavky.....	- 12 -
3.1.4 P – výkonnostní požadavky.....	- 12 -
3.1.5 S – požadavky na udržovatelnost softwaru.....	- 13 -
3.2 Scénáře případu užití (use-case scenario).....	- 13 -
3.2.1 Systém z pohledu uživatele a admina.....	- 17 -
3.3 Návrh uživatelského rozhraní.....	- 18 -
4 Návrh architektury.....	- 20 -
5 Statický náhled na systém.....	- 23 -
6 Dynamický náhled na systém.....	- 24 -
7 Prvky architektury.....	- 25 -
8 Implementace.....	- 27 -
9 SEO optimalizace.....	- 30 -
10 Závěr.....	- 32 -
11 Seznam použité literatury.....	- 33 -
Přílohy na CD/DVD.....	- 34 -

Úvod

V této bakalářské práci je vytvářen systém pro zobrazování reklam a jejich správu.

Chtěl jsem si navrhnout a vytvořit vlastní systém pro správu reklamy. Systém má několik pilířů. Uživatelům chci nabídnout službu tvorby blogu. Cílovou skupinou budou malí živnostníci, jako například instalatéři, zedníci, malíři apod. Tento blog bude v rámci systému uživatelům vytvářen za snížený poplatek, protože se na blogu bude zobrazovat reklama, kterou zvolí administrátor systému. Prozatím máme jednu šablonu blogu s rozmístěním prvků webu, ovšem v budoucnu bude systém nabízet šablon mnohem více. Architektura systému tedy musí podporovat snadné napojení na systém reklamy, což je v této bakalářské práci řešeno n-vrstvou architekturou. Neboli v mém případě tří-vrstvou architekturou a zejména pak business vrstvou.

Druhým pilířem systému je tvorba podpůrných systémů pro zadávání reklamy. Když budeme mít systémem vytvořeno dostatek blogů na kterých bude dostatečná měsíční návštěvnost, tak poté bude systém nabízet zadávání reklamy na blozích uživatelů. Z toho vyplývá, že si uživatel bude moci zvolit, jakou reklamu bude na svém blogu zobrazovat.

V průběhu tvorby prochází systém několika fázemi. První fáze, nazvaná analýza, je nezbytná pro teoretickou část práce. Pro definování softwarových požadavků, které určují vlastnosti výsledného systému, se postupuje podle osvědčených metodik. Podle jedné z těchto metodik, konkrétně metodiky FURPS se člení požadavky na redakční systém v této bakalářské práci.

Krátce si popíšeme, jaké typy reklam bude systém nabízet. První typ je reklama na určitou dobu, jejíž cena se bude počítat následujícím způsobem. Stanovíme určitou cenu za jeden den, a tu poté vynásobíme počtem dnů. Druhý typ se nazývá reklama na určitou částku. Reklama bude mít stanovenou pevnou částku a ta se bude postupně odečítat. Uživatelské rozhraní systému bude realizováno pomocí webové stránky. Systém bude možné spouštět skrze přenosná zařízení s připojením k internetu, jako jsou například mobilní telefony, tablety a další.

Práci jsem rozdělil do několika sekcí, mám-li jmenovat hlavní z nich, jedná se o analýzu, návrh, implementaci a závěr. V sekci analýzy řeším problematiku sběru požadavků. Dále zde uvádím příklady řešených problémů např. v podobě FURPS požadavků nebo scénářů případů užití. V sekci implementace nejprve volím vyhovující architekturu, tu poté popíšu a zdokumentuji. Poslední sekce je závěr, v němž uvádím shrnutí použitých materiálů a také problematiky řešené v této bakalářské práci.

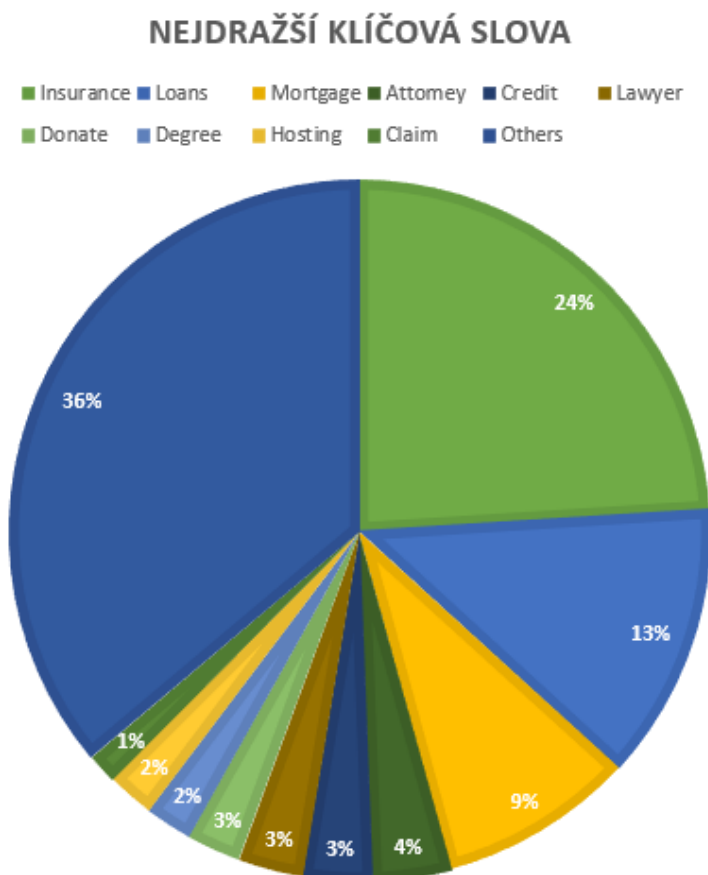
1 Rozbor stávajících systémů

Systémů pro inzerci reklam na českém internetu existuje několik druhů.

Prvním a nejznámějším je Google AdWords, který nabízí hned několik způsobů inzerce reklamy. Jmenuji například platbu za prokliknutí neboli PPC (pay per click), platbu za počet zobrazení čili PPV (pay per view). Reklama v Google AdWords se může zobrazovat ve výsledcích vyhledávání Google (případně dalších partnerských vyhledávačů) nebo na partnerských webech. Ze všech PPC systémů má asi nejpropracovanější možnosti nastavení a cílení kampaní.

Minimální cena za proklik u Google AdWords činí 0,10 Kč. Nejdražší proklik je ve Spojených arabských emirátech, kde se cena za proklik pohybuje v průměru o 8% více, než je tomu v USA. (1) Dále nejvyšší cenu za proklik kromě Spojených arabských emirátů a USA drží, země jako Rakousko, Austrálie, Brazílie, Velká Británie, Nový Zéland, Chile, Švýcarsko, Itálie, Kanada a Německo. Všechny tyto státy mají silné ekonomické vztahy s USA, nebo vysoce stoupající ekonomiku a také velkou populaci obyvatelstva, které hovoří anglickým jazykem. Tento fakt je dán také tím, že vyhledávač Google je v těchto zemích monopol a tím pádem nemá konkurenci ostatních vyhledávačů.

Na tomto grafu si můžeme prohlédnout nejdražší hledaná slova na Googlu (2).



Obrázek č. 1 – Graf nejdražších klíčových slov

Jako dalším známý PPC systémem uvedu Sklik provozovaný portálem Seznam.cz. Reklama se zde zobrazuje ve výsledcích vyhledávání na Seznam.cz, kromě toho také v Seznam zboží, Encyklopedii, katalogu firem a Sbazaru. Systém Sklik je uživatelsky velmi podobný systému AdWords.

Zde minimální cena za proklik činí 0,20 Kč.

Na obrázku číslo 2. můžeme vidět průměrnou cenu pro klíčové slovo *modely aut*. Ta se pohybuje mezi 4,30 Kč až 0,97 Kč (3).

Klíčové slovo	Hledanost	Roční trend	Konkurence	Cena Ø
rc modely aut	7 772			4,32 Kč
modely aut	2 828			2,40 Kč
kovové modely aut	1 866			1,80 Kč
modely aut škoda	272			3,35 Kč
modely aut 1 43	171			2,31 Kč
modely aut 1:18	165			2,12 Kč
modely aut 1:43	156			2,49 Kč
modely aut abrex	138			1,76 Kč
plastikové modely aut	134			1,81 Kč
kovove modely aut	122			1,59 Kč
model aut	120			2,42 Kč
modely aut 1 18	119			1,85 Kč
sběratelské modely aut	112			0,97 Kč
rc modely aut 1:10	104			3,63 Kč
kovové modely aut 1:43	86			1,41 Kč
modely aut na dálkové ovládání	82			2,23 Kč
modely aut tatra	81			1,93 Kč
kovové modely aut 1 43	76			-

Obrázek č. 2 - Průměrné ceny klíčových slov PPC systému Sklik

Nesmím opomenout ani další český PPC systémem, eTarget, který provozuje portál Atlas.cz. Reklama se v tomto případě zobrazuje ve výsledcích vyhledávání portálu Atlas.

V systému eTarget je cena za proklik stanovena na 1 Kč.

Cílem této bakalářské práce, je vytvořit přehledné uživatelské rozhraní, aby uživatel mohl pohodlně spravovat své reklamní kampaně. Tento redakční systém se inspiroje výše zmíněnými PPC systémy, nabízí tři druhy reklamních kampaní, uvedených v úvodu této práce.

Google AdWords

1 Informace o vaší firmě 2 Vaše první kampaň 3 Fakturace 4 Kontrola

Bezplatná podpora: 800 720 0
Další mezinárodní telefonní čísla

Vítejte ve službě Google AdWords!

Pomůžeme vám zaregistrovat se a začít inzerovat v několika krocích.
Máte již zkušenosti s AdWords? [Přeskočte asistované nastavení.](#)

Jakou máte e-mailovou adresu?

email@example.cz

▲ Zadejte e-mailovou adresu do pole výše.

Jaký máte web?

www.example.cz/knihy

☐ Ano, chci dostávat personalizované tipy a doporučení, jak zlepšit výkon reklam.

☐ Ne, nechci dostávat personalizované tipy a doporučení, jak zlepšit výkon reklam.

Pokračovat

Obrázek č. 3 - PPC systém Google AdWords

Google AdWords

Bocek.J@gmail.com

1 Informace o vaší firmě 2 Vaše první kampaň 3 Fakturace 4 Kontrola

Bezplatná podpora: 800 720 0
Další mezinárodní telefonní čísla

Vaše první kampaň

Kampaň se zaměřuje na téma nebo skupinu produktů. Při vytváření kampaně je třeba nastavit rozpočet, zvolit publikum a vytvořit reklamu. Za výběr možností vám nebudou účtovány žádné poplatky a nastavení můžete později kdykoli změnit.

1. Rozhodněte se, kolik chcete utratit

Váš rozpočet 10,00 Kč na den

2. Zvolte cílové publikum

Oblasti Česká republika

Sítě Vyhledávací síť, Obsahová síť

Klíčová slova reklama ppc reklama intenciová reklama

3. Nastavte nabídku


Nabídka Automaticky nastavit nabídky s cílem zajistit co nejvíce kliknutí v rámci rozpočtu

Denní potenciální zásah
Pouze Vyhledávací síť

19+ kliknutí
2K+ zobrazení


Obrázek č. 4 PPC systém Google AdWords zadání kampaně

3. Nastavte nabídku

Nabídka Automaticky nastavit nabídky s cílem zajistit co nejvíce kliknutí v rámci rozpočtu 

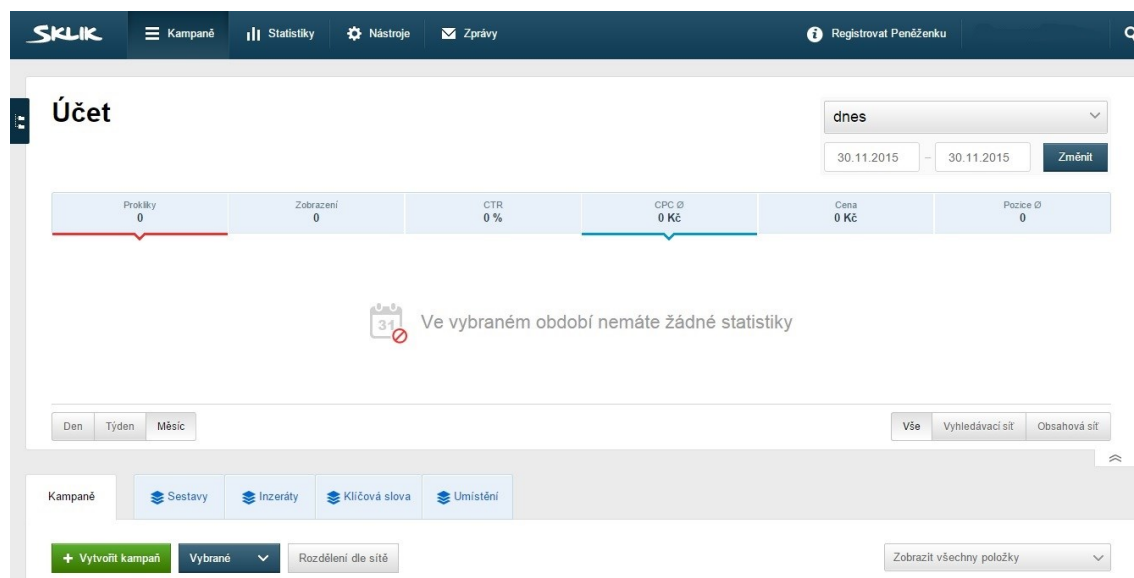
4. Vytvořte reklamu

Textová reklama

Reklama
www.stranka.cz
internetová reklama
pokračování internetové reklamy 

[Uložit a pokračovat](#)

Obrázek č. 5 PPC systém Google AdWords pokračování zadání kampaně



Obrázek č. 6 PPC systém Sklik

Vytvoření nové kampaně

1. Označení a nastavení cen

Název kampaně:

Denní rozpočet: Kč

Název sestavy:

Výchozí cena za proklik: Kč

2. Vytvoření inzerátu nebo banneru

Vyplnit inzerát:

Titulek: Zbývá 18 znaků

Popisek 1: Zbývá 21 znaků

Popisek 2: Zbývá 27 znaků

Viditelné URL: Zbývá 27 znaků

Cílové URL: Zbývá 929 znaků

Náhled inzerátu

[Reklama](#)
Detail reklamy
Reklama2
Firmy.cz

3. Zadání klíčových slov

Obrázek č. 7 PPC systém Sklik vytvoření kampaně (nastavení denního rozpočtu, ceny za proklik a dále pak tvorba banneru)

3. Zadání klíčových slov

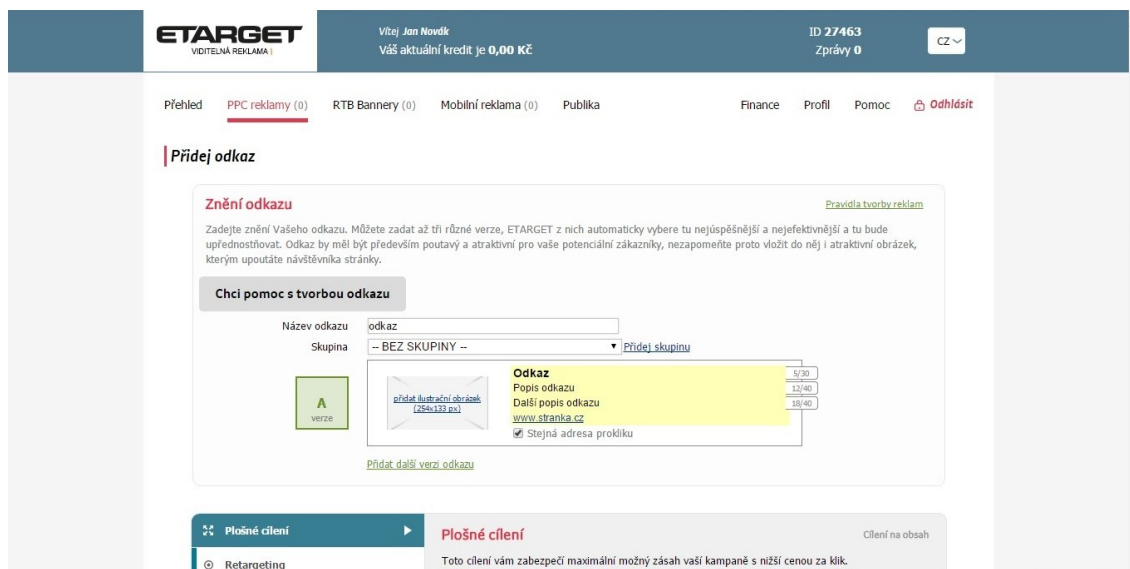
reklama ☐ Komerčně příbuzná

Klíčové slovo	Hledanost	Roční trend	Konkurence	Cena Ø
hračky z tv reklamy	210			2,65 Kč
reklama na kofolu prasátko	161			-
ppc reklama	1 538			76,15 Kč
reklama	1 688			4,79 Kč
reklama na kofolu	898			-
světelná reklama	1 064			40,66 Kč
hračky z reklamy	68			1,82 Kč
kofole reklamy	78			-
reklama na internetu	1 024			15,70 Kč
pavouk z reklamy	80			1,89 Kč
světelná reklamy	581			3,00 Kč
výroba světelné reklamy	676			20,05 Kč
venkovní reklama	474			10,60 Kč
3d reklama	458			-
internetová reklama	757			4,60 Kč
reklamy	407			3,86 Kč

Klíčová slova

- [ppc reklama]
- "ppc reklama"
- ppc reklama
- [reklama]
- "reklama"
- reklama
- [reklama na internetu]
- "reklama na internetu"
- reklama na internetu
- [ppc reklamy]
- "ppc reklamy"
- ppc reklamy
- [reklamy]
- "reklamy"
- reklamy

Obrázek č. 8 PPC systém Sklik pokračování vytvoření kampaně (zadávání klíčových slov)



Obrázek č. 9 PPC systém eTarget vytvoření odkazu (tvorba odkazu zdáváme jeho název a zařazení do skupiny)

Srovnávací tabulka všech tří PPC systémů

	Google AdWords	Sklik	eTarget
vzhled	8/10	7/10	5/10
přehlednost	9/10	8/10	6/10
cena	9/10	8/10	7/10
celkový dojem	9/10	8/10	6/10

V systému Google AdWords se mi velmi líbí přívětivé uživatelské rozhraní a velice snadná tvorba reklamy a její následná správa. Systém Sklik, již tolik nevyniká uživatelským rozhraním, ale zadávání reklamní kampaně je velice přehledné a intuitivní. Poslední systém eTarget mě nezaujal ani svým uživatelským rozhraním ani tvorbou reklamní kampaně. Vše je dosti nepřehledné. Systém hodnocení, který zde uvádím je čistě můj subjektivní názor.

Z tabulky pak můžeme vidět, že nejlépe je na tom systém Google AdWords, dále pak Sklik a nejhůř z hodnocení vyšel PPC systém eTarget. V mém systému se, proto budu nejvíce inspirovat a také snažit se přiblížit systémům Google AdWords a Sklik.

2 Postup vypracování

Pro dodržení správného vývoje softwaru se zavádí takzvaný softwarový proces. K vypracování této práce používám softwarový proces Rational Unified Process (RUP), který následně krátce popisuji.

RUP je objektově orientovaný iterativní přístup k životnímu cyklu systému. Tento proces si za cíl klade zaručit produkci kvalitního softwaru, který splňuje požadavky klienta, s předpověditelným časovým plánem a rozpočtem. Stěžejní fáze RUP jsou podle (4) (5) tyto:

Zahájení (Inception)

Ve fázi zvané zahájení musí být ustanoveny používané prostředky. Například pro práci s analýzou použijeme Unified Modeling Language (UML) (6) (7) (8). Dále je třeba definovat rozsah vytvářeného softwaru, pro který určíme všechny vnější prostředky, se kterými bude software pracovat, například uživatele nebo jiné systémy, a definujeme tyto prostředky. To zahrnuje i identifikaci podnikových procesů a jejich popis za pomoci scénářů případů užití.

Výsledkem této fáze bude dokument pojednávající například o potřebách softwaru, jeho hlavních funkcích, či počátečních modelech případu užití.

Příprava (Elaboration)

Fáze příprava si za cíl klade analýzu doménové problematiky, ustanovení architektury, vytvoření softwarového plánu a odstranění těch částí softwarového návrhu, které považujeme za vysoce rizikové. K úspěšnému dokončení této fáze, musíme mít tzv. kompletní přehled o systému, přestože v této situaci to není zcela možné, kvůli rozsahu softwaru. Architektonická rozhodnutí musíme vytvořit s porozuměním vzhledem k celému systému, neboli jak k jeho rozsahu, tak k funkčním či nefunkčním požadavkům. Výsledek této fáze rozhoduje o dalším možném postupu, který však nemusí nastat.

Na konci této fáze získáváme model případu užití, spolu se všemi požadavky, účastníky, popisem softwarové architektury, spustitelným modelem architektury a souhrnem všech rizik atd. Všechny tyto požadavky máme kompletně zdokumentovány.

Vypracování (Construction)

Při vypracování implementujeme ostatní, zbývající komponenty a funkce, které má požadovaný produkt poskytovat. Samotné vypracování probíhá ve fázích. Každá iterace by měla být zakončena vznikem funkční části (jakýchsi předběžných ukázek). Tyto předběžné ukázky předvedeme zákazníkovi nebo vedoucímu vývoje softwaru. Poté konzultujeme případné změny, a také nedostatky.

Na konci fáze je potřeba rozhodnout, zda vytvořený software můžeme nasadit. Musíme však zvážit jeho připravenost vůči možným rizikům. Ve většině případů je takovýto software nasazen jako tzv. beta verze.

Nasazení (Transition)

Účelem této fáze je nasadit výsledný produkt jeho koncovým uživatelům. Po nasazení koncovým uživatelům začíná fáze testování, ve které dochází k identifikaci problémů, které případně mohou vynutit vytvoření nové verze, a také k dílčí aktualizaci programu. Jedná se o poměrně časově náročnou fázi. Její časová náročnost závisí na komplexnosti vytvářeného softwaru.

V tomto konečném bodě se rozhodne, jak dobře byly splněny požadavky na daný software. Pokud daný software neodpovídá požadavkům, je třeba vrátit se k fázi *zahájení*.

Použití RUP v rámci bakalářské práce:

Jako předloha k vypracování této práce slouží obrázek č. 10, který rozděluje softwarový proces na jednotlivé fáze, skládající se z iterací.

Popis obrázku č. 10:

Osa horizontální zde ukazuje jednotlivé iterace daných fází. Fáze jsou rozděleny na iterace, po kterých vzniká prototyp produktu, konzultovaný s vedoucím práce. Oblast práce, postupně se přesouvající z jedné fáze do druhé, závisí na času, který jsme nad vývojem softwaru strávili.



Obrázek č. 10 Fáze projektu

3 Analýza a návrh

Analýza projektu nám slouží k tomu, abychom si stanovili požadavky, které budeme klást na systém. Výstupem pak bude seznam požadavků podle metody FURPS a scénáře případu užití v systému.

Návrh nám slouží k tomu, abychom si mohli představit, jak systém bude vypadat graficky.

Výstupem pak jsou jednotlivé formuláře nebo tabulky v systému.

3.1 Správa požadavků na systém podle metody FURPS

V této podkapitole definuji metoda FURPS, a uvedu důvody, proč ji používat. Původním tvůrcem metody FURPS byl Robert Grady, poté však byla rozšířena společností Rational Software. Metoda FURPS rozděluje softwarové požadavky do 5 kategorií. Tyto požadavky jsou nezávisle ohodnoceny podle měřítka vytvořeného pro daný software. Kategorie pro rozdělení požadavků jsou popsány podle (9).

Následujících pět částí metody FURPS znázorňuje oblasti, do kterých je systém rozdělen.

Functionality (funkčnost): Funkční požadavky jsou považovány za architektonicky významné. Každý z těchto požadavků představuje funkci systému.

Usability (použitelnost): Požadavky založené na otázkách uživatelského rozhraní, například dostupnost, konzistence, estetika.

Reliability (spolehlivost): Zahrnuje aspekty, jako je dostupnost, přesnost a využitelnost.

Performance (výkon): Patří zde propustnost, doba odezvy systému, doba zotavení a doba spouštění.

Supportability (podpora): Obsahuje řadu dalších požadavků, mezi něž řadíme snadné testování, přizpůsobivost, udržitelnost, kompatibilita, konfigurovatelnost, škálovatelnost a další.

Níže uvádím šablonu pro zápis požadavků, a zároveň příklady požadavků na systém vytvářený v této bakalářské práci. Šablona bude implementována jako šablona základní. Tu lze v systému dále rozšiřovat. Poté následuje ukázka výčtu požadavků systému.

Všechny požadavky jsou uvedeny v příloze.

Šablona softwarového požadavku

Id: jednoznačný identifikátor požadavku

Název: název požadavku

Popis: detailní popis průběhu požadavku

Priorita: priorita požadavku na stupnici 1-10 (1- nejnižší, 10- nejvyšší)

Závislost: závislost na jiných požadavcích

3.1.1 F – funkční požadavky

Id: FR 1

Název: Zákazník se může přihlásit do systému

Popis:

Priorita: 10

Závislost:

Id: FR 2

Název: Požadavek na změnu reklamy

Popis: Zákazník pošle adminovi email, kterým ho informuje o požadované změně.

Priorita: 4

Závislost: FR 1

Id: FR 3

Název: Požadavek na smazání reklamy

Popis: Zákazník pošle adminovi email, kterým ho informuje o žádosti ke smazání reklamy.

Priorita: 5

Závislost: FR 1

Id: FR 4

Název: Požadavek na smazání účtu zákazníka

Popis: Zákazník pošle adminovi email, kterým ho informuje o žádosti k smazání svého účtu

Priorita: 5

Závislost: FR 1

Id: FR 5

Název: Zákazník může mít registrováno více reklam

Popis: Zákazník musí mít možnost registrace více reklamních kampaní

Priorita: 6

Závislost: FR 1

3.1.2 U – požadavky vhodnosti k použití

Id: UR 1

Název: Uživatelsky přívětivý systém

Popis: Vzhled systému a menu musí být natolik přehledný, aby se uživatel v systému ihned orientoval

Priorita: 8

Závislost:

Id: UR 2

Název: Objednání reklamy bez nutnosti dokumentace

Popis: Systém musí být natolik přehledný, aby si zákazník dokázal objednat reklamu bez použití dokumentace

Priorita: 7

Závislost:

3.1.3 R – spolehlivostní požadavky

Id: RR 1

Název: Maximálně dva výpadky systému

Popis: Systém může mít maximálně dvakrát do roka výpadek na dvě hodiny

Priorita: 9

Závislost:

Id: RR 2

Název: Uvedení systému do původního stavu

Popis: Po pádu systému se systém uvede do původního stavu

Priorita: 9

Závislost:

3.1.4 P – výkonnostní požadavky

Id: PR 1

Název: Schopnost provozu systému

Popis: Systém musí být schopen běžet nepřetržitě 22 hodin denně a 7 dní v týdnu

Priorita: 9

Závislost:

Id: PR 2

Název: Zobrazení nových reklam

Popis: Systém by měl být schopen do 10 sekund zobrazit nově objednané reklamy na 1000 stránkách

Priorita: 8

Závislost:

3.1.5 S – požadavky na udržitelnost softwaru

Id: SR 1

Název: Technická podpora systému

Popis: Technická podpora systému by měla dostupná nepřetržitě

Priorita: 9

Závislost:

Id: SR 2

Název: Chyby systému

Popis: Jakékoli chyby systému jsou okamžitě zasílány emailem adminovi

Priorita: 8

Závislost:

3.2 Scénáře případu užití (use-case scenario)

Další bod analýzy se skládá z rozdělení systémových funkcí a přidělení těchto funkcí k jejich „aktérům“. Jako hlavní aktéři vystupují v této práci zákazník a admin. K vizualizaci vazeb mezi uživatelem a funkcemi slouží diagram použití, který uvádím níže na obrázku č. 12. Diagram případů užití (obrázek č. 12) ukazuje, jaké funkce systém obsahuje, a který uživatel má k daným funkcím přístup. Dá se z něj tedy vyčíst název funkce, nikoliv však její přímá funkčnost. K tomu slouží scénář případu užití, který nemá pevně danou podobu, ale může být realizován ve formě tabulky nebo prostého textu. Scénář případu užití doplňuje diagram případu užití a popisuje plný průběh dané funkce.

Šablona scénáře případu užití

ID	Zkratka Use-case (UC) a jedinečné číslo
Název	Krátký název, který popisuje daný scénář
Související požadavky	Požadavky, které musí být splněny, aby scénář mohl proběhnout
Předpoklady	Předpoklady, které musí být splněny před započítím scénáře
Úspěšný konečný stav	Stav, ve kterém se nachází scénář po úspěšném průběhu
Neúspěšný konečný stav	Stav, ve kterém se nachází scénář po neúspěšném průběhu
Primární aktéři	Hlavní aktér pracující se scénářem

Další aktéři	Další aktéři pracující se scénářem
Hlavní průběh	Hlavní linie průběhu scénáře
Rozšiřující průběh	Linie průběhu scénáře v případě chyby
Alternativní průběh	Alternativní linie průběhu rozšiřující hlavní průběh scénáře

Příklad složitějšího scénáře případu užití

ID	UC2
Název	Objednání reklamy
Související požadavky	Zákazník musí být přihlášen do systému
Předpoklady	Zákazník je přihlášen do systému a je na stránce pro objednávání reklamy
Úspěšný konečný stav	Zákazník si objednal reklamu
Neúspěšný konečný stav	Objednávka není evidována- systém se vrátil do původního stavu
Primární aktéři	Zákazník
Další aktéři	
Hlavní průběh	<ol style="list-style-type: none">1. Zákazník se nachází na stránce pro objednání reklamy2. Systém zobrazí formulář pro zadání parametrů objednávky3. Zákazník zadá povinné údaje a formulář potvrdí (viz tabulka povinných údajů)4. Systém zkontroluje zadané údaje a zobrazí formulář pro zadání doplňujících údajů5. Zákazník zadá doplňující údaje6. Systém zkontroluje zadané doplňující údaje7. Zákazník si zvolí typ reklamy8. Zákazník potvrdí uložení zvolené reklamy9. Systém zkontroluje zadané údaje10. Systém uloží objednávku reklamy a informuje zákazníka11. Zákazníkovi je zpřístupněna objednaná reklama
Rozšiřující průběh	<ol style="list-style-type: none">4. Nebyly zadány všechny povinné údaje<ol style="list-style-type: none">4.1 Systém se zákazníka zeptá, jestli chce doplnit chybějící povinné údaje a pokud zákazník zvolí ano, je vrácen k bodu 3 hlavního průběhu4.2 Pokud zákazník zvolí ne, je jeho objednávka reklamy stornována

Alternativní průběh	7. Zákazník si zvolí typ reklamy 7.1 Reklama na určitou dobu 7.2 Reklama na určitou částku 7.3 Reklama na určitý počet kliků
---------------------	---

Další příklad složitějšího scénáře případu užití

ID	UC3
Název	Úprava reklamy
Související požadavky	Admin musí být přihlášen do systému
Předpoklady	Admin je přihlášen do systému a je na stránce pro vyhledání zákazníka a reklamy
Úspěšný konečný stav	Admin upravil reklamu
Neúspěšný konečný stav	Systém se vrátil do původního stavu
Primární aktéři	Admin
Další aktéři	
Hlavní průběh	1. Systém zobrazí stránku pro vyhledání zákazníka 2. Admin vybere zákazníka 3. Systém zobrazí všechny reklamy daného zákazníka 4. Admin vybere konkrétní reklamu a upraví údaje reklamy 5. Admin potvrdí uložení reklamy 6. Systém zkontroluje zadané údaje 7. Systém uloží upravenou reklamu 8. Reklama je aktualizována v systému
Rozšiřující průběh	6. Nebyly zadány všechny povinné údaje 6.1 Systém se admina zeptá, jestli chce doplnit chybějící údaje reklamy a pokud admin zvolí ano, je vrácen k bodu 4 hlavního průběhu 6.2 Pokud admin zvolí ne, je úprava reklamy stornována
Alternativní průběh	

Třetí příklad složitějšího scénáře případu užití

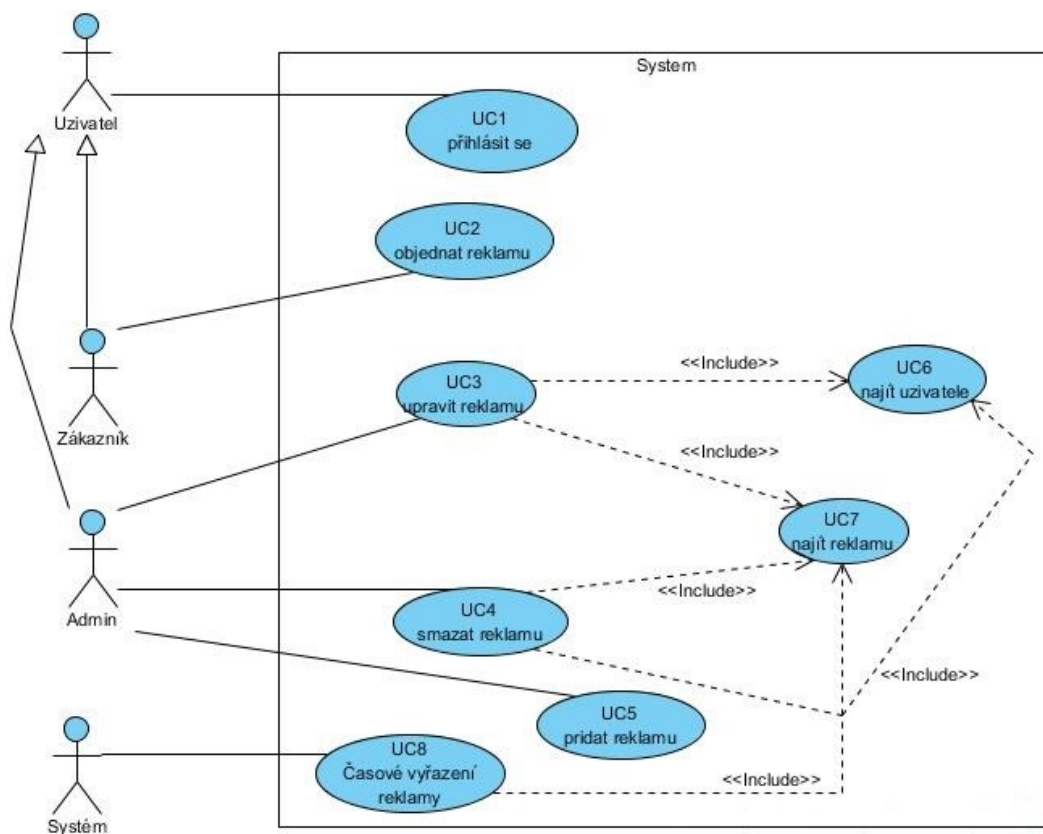
ID	UC4
Název	Smazání reklamy
Související požadavky	Admin musí být přihlášen do systému
Předpoklady	Admin je přihlášen do systému a je na stránce pro vyhledání zákazníka a reklamy
Úspěšný konečný stav	Admin smazal reklamu
Neúspěšný konečný stav	Systém se vrátil do původního stavu
Primární aktéři	Admin
Další aktéři	
Hlavní průběh	1. Systém zobrazí stránku pro vyhledání zákazníka 2. Admin vybere zákazníka 3. Systém zobrazí všechny reklamy daného zákazníka 4. Admin vybere konkrétní reklamu a zadá smazání reklamy 5. Admin potvrdí smazání reklamy 6. Systém zkontroluje zadané údaje 7. Systém smaže danou reklamu 8. Reklama je ze systému smazána
Rozšiřující průběh	6. Nastala chyba 6.1 Smazání reklamy je stornováno
Alternativní průběh	

Tabulka povinných údajů

Jméno	varchar(30)
Příjmení	varchar(30)
Telefon	integer
Ulice a č.p.	varchar(30)
Město	varchar(30)
PSČ	integer
Email	varchar(30)
Heslo	varchar(30)

3.2.1 Systém z pohledu uživatele a admina

Tento diagram znázorňuje chování systému z pohledu uživatele a admina. Jeho účelem je zobrazit jednoduchý náhled nad funkcemi systému. Na tomto konkrétním obrázku č. 11 lze vidět, že zákazník i admin mají přístup k scénáři UC1 neboli přihlášení. Na obrázku dále vidíme, že zákazník má přístup k objednávce reklamy. Admin má pak přístup ke scénáři UC3, který nese název *upravit reklamu*, dále pak ke scénáři *smazat reklamu* a *přidat reklamu*. Ve scénářích *upravit reklamu* i *smazat reklamu* můžeme vidět, že obsahují další scénáře UC6 *najít uživatele* a UC7 *najít reklamu*.

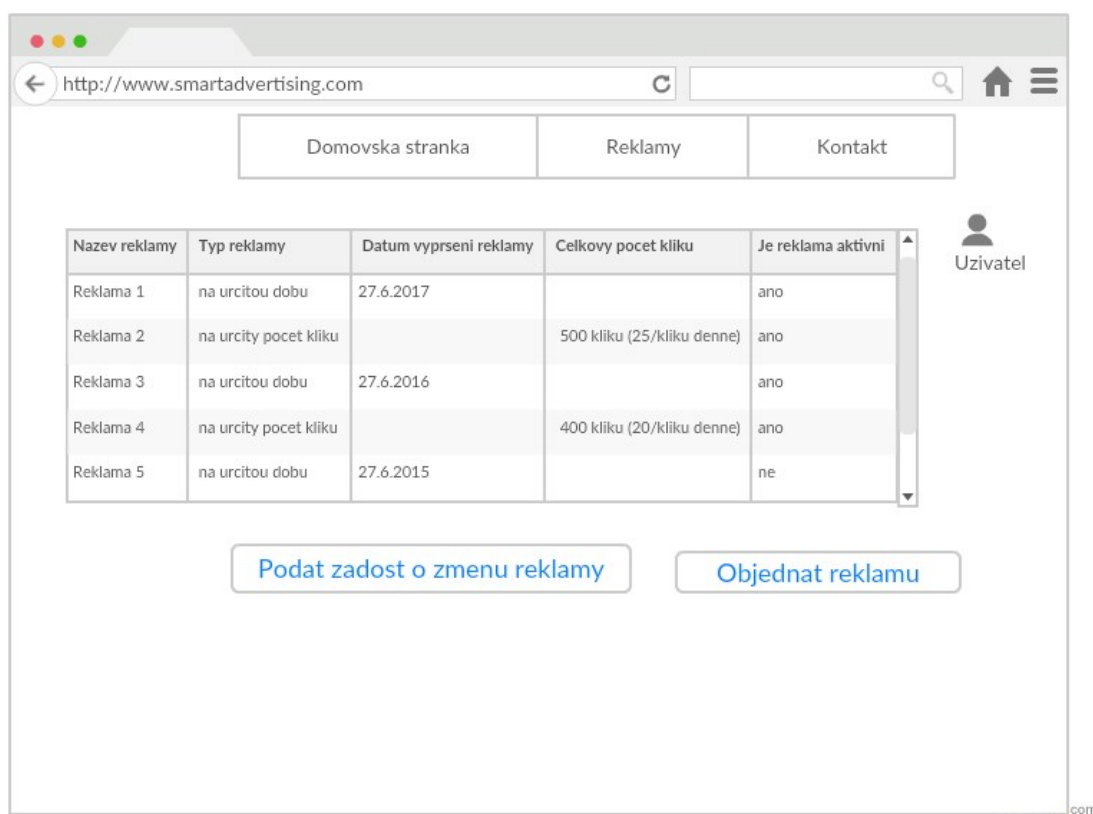


Obrázek č. 11 Use-case diagram

3.3 Návrh uživatelského rozhraní

Pro znázornění vzhledu systému se vytváří prototyp uživatelského rozhraní. Tento prototyp slouží ke znázornění budoucího vzhledu systému zákazníkovi, tak aby mohl vznést námitky nebo upřesnit požadavky, které na budoucí vzhled systému klade. Pomáhá při vypracovávání konečného vzhledu systému, není však definitivní, jde pouze o předběžný návrh rozmístění komponent na viditelné části systému.

Obrázek č. 12 znázorňuje, jak by měl vypadat seznam reklam uživatele. V dolní části obrázku vidíme tlačítka pro objednávání reklamy a pro odeslání žádosti ke změně reklamy. Ve vrchní části je umístěno hlavní menu pro pohyb na stránkách. Vpravo se pak nachází účet zákazníka.



Obrázek č. 12 Seznam reklam zákazníka

Na obrázku č. 13 nacházíme formulář pro objednání reklamy. Rozložení prvků na stránce se podobá předchozímu obrázku, aby uživatelské prostředí bylo jednoduché a přehledné.

The screenshot shows a web browser window with the address bar displaying `http://www.smartadvertising.com`. The browser has standard navigation buttons (back, forward, refresh) and a search icon. Below the address bar is a navigation menu with three items: "Domovska stranka", "Reklamy", and "Kontakt". On the right side of the page, there is a user icon and the text "Uzivatel". The main content area contains a form for ordering an advertisement with the following fields:

- Název reklamy: Text input field.
- Typ reklamy: Dropdown menu with "Typ reklamy" selected.
- Klíčová slova: Text input field.
- URL reklamy: Text input field.
- Popis reklamy: Text input field.

At the bottom of the form, there are two buttons: "Potvrdit" (Confirm) and "Storno" (Cancel).

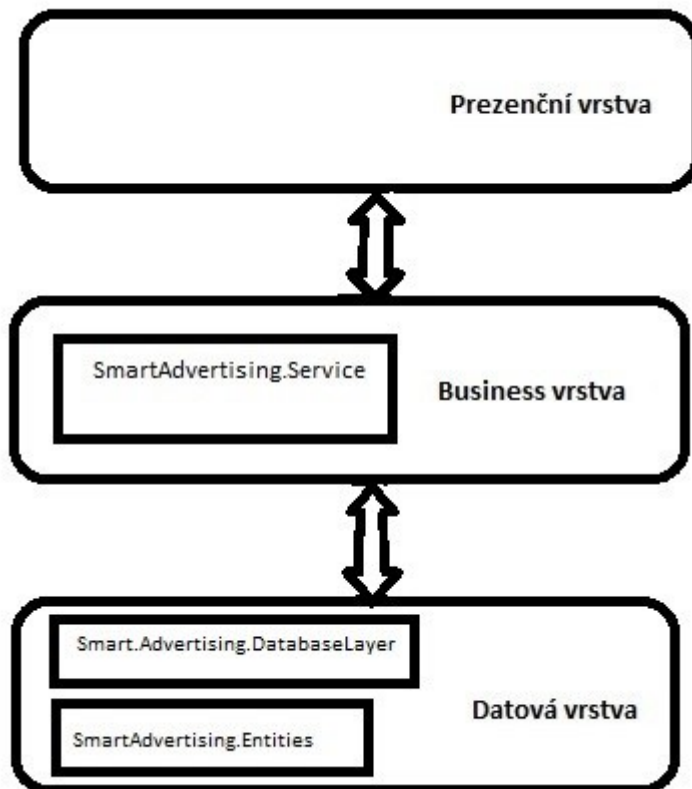
Obrázek č. 13 Objednávka reklamy

4 Návrh architektury

Vytváříme systém, který se bude vyvíjet. Postupem času přibudou tisíce uživatelů, budeme muset zvětšit tým programátorů. Systém bude nutné pokrýt testy.

Proto jsem se rozhodoval mezi dvouvrstvou a třívrstvou architekturou. Dvouvrstvou architekturu jsem nakonec nepoužil z toho důvodu, že business vrstva i datová vrstva jsou spojené do jediné vrstvy. Dvouvrstvá architektura také nezvládá velký počet uživatelů má mnohem menší zabezpečení a obtížně se škáluje. Rozhodl jsem se tedy pro třívrstvou architekturu, která se jeví jako nejlepší možné řešení. A to z důvodů dobrého zabezpečení, pojmutí velkého počtu uživatelů a kontroly nad jednotlivými vrstvami a jejich případné budoucí rozšíření.

Třívrstvá architektura je jedním ze základních vzorů softwarého vývoje (10) (11).



Obrázek č. 14 Třívrstvá architektura

Je to softwarová architektura, která odděluje prezenční vrstvu, business vrstvu a vrstvu datovou.

Jednotlivé vrstvy plní samostatné úkoly a je možné je vyvíjet, udržovat a měnit nezávisle za předpokladu zachování neměnného (zpětně kompatibilního) rozhraní mezi jednotlivými vrstvami.

Rozdělení vrstev přináší některé zásadní výhody:

1. Možnost nezávislé správy jednotlivých vrstev:

Při zachování integračních vazeb je možné provádět i zásadní změny v jednotlivých vrstvách bez nutnosti zásahu do vrstev dalších.

2. Dimenzování:

Jednotlivé vrstvy je možné uzpůsobit požadavkům na jejich výkonnost bez ohledu na vrstvy další.

3. Bezpečnostní nastavení:

Jednotlivé vrstvy lze přizpůsobit jejich použití a požadavkům na ně.

4. Oddělení znalostí:

Správné rozdělení jednotlivých vrstev umožňuje oddělit znalosti nutné pro jejich správu a alokovat na ně vhodné zdroje bez zbytečných překryvů a s ohledem na složitost jednotlivých

5. Oddělení kompetencí:

Jasně oddělení kompetencí v případě zapojení více dodavatelů. Jednotlivé vrstvy a rozhraní mezi nimi lze s výhodou využít jako hranice kompetencí mezi dodavateli jednotlivých částí řešení.

Prezentační vrstva:

Neboli vrstva uživatelského rozhraní je odpovědná za interakci s obsluhou – sběr informací a naopak jejich zobrazení v uživatelsky čitelné a přívětivé podobě.

Nejčastěji samozřejmě v podobě výstupního HTML/CSS/JavaScript kódu.

Podporuje uživatelské interakce v podobě vkládání dat a jejich zobrazování.

Musí obsahovat bezpečnostní prvky a opatření zajišťující fungování projektu v prostředí internetu.

Měla by také zahrnovat funkcionalitu z hlediska SEO a dalších on-line marketingových aktivit.

Business vrstva:

Je odpovědná za přenos informací mezi prezenční a datovou vrstvou a to včetně začlenění potřebné rozhodovací logiky nebo transformací předávaných dat (výpočty, agregace informací z více datových zdrojů atp.).

Zajišťuje odstínění prezentační vrstvy od datové vrstvy a poskytuje všechny potřebné funkce pro prezenční vrstvu.

Dočasné úložiště informací, které je nutné pro využití v prezentační vrstvě a technicky nelze zajistit jejich dostupnost přímo z datové vrstvy.

Datová vrstva:

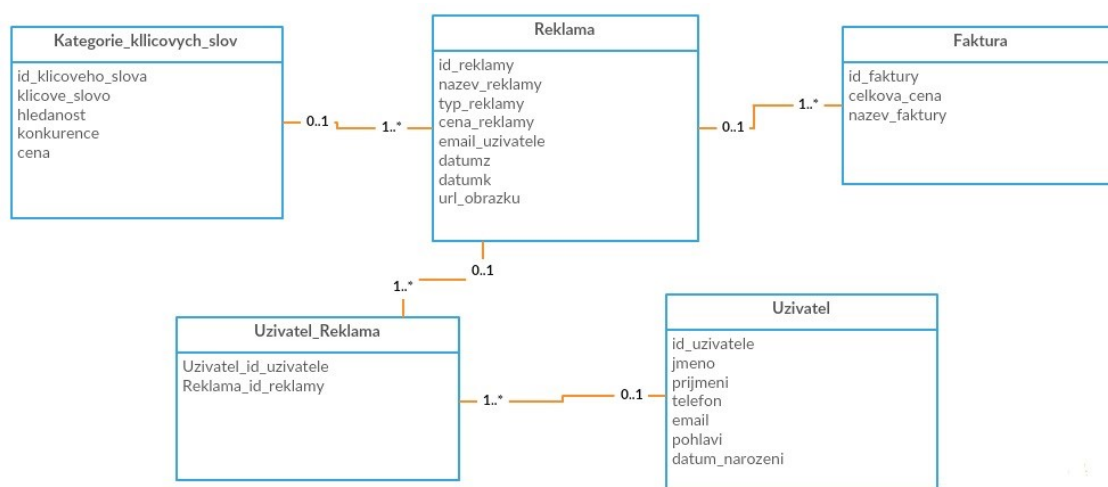
Hlavním úkolem této vrstvy je poskytovat bezpečné a spolehlivé úložiště dat informačního systému.

Poskytuje základní zabezpečení přístupu k datům a prostředky pro rychlý přístup k datům a jejich vyhledávání.

Datová vrstva také přímo komunikuje s business vrstvou a zajišťuje základní integritu dat.

5 Statický náhled na systém

Diagram tříd představuje statický pohled na modelovaný systém a jeho úkolem je znázornit typy objektů v systému a jejich vztahy. Základem systému je tabulka Reklama, kde ukládáme jednotlivé reklamy. Tabulka Kategorie_klicovych_slov pak určuje zaměření reklamy (např. cestování, volný čas apod.)



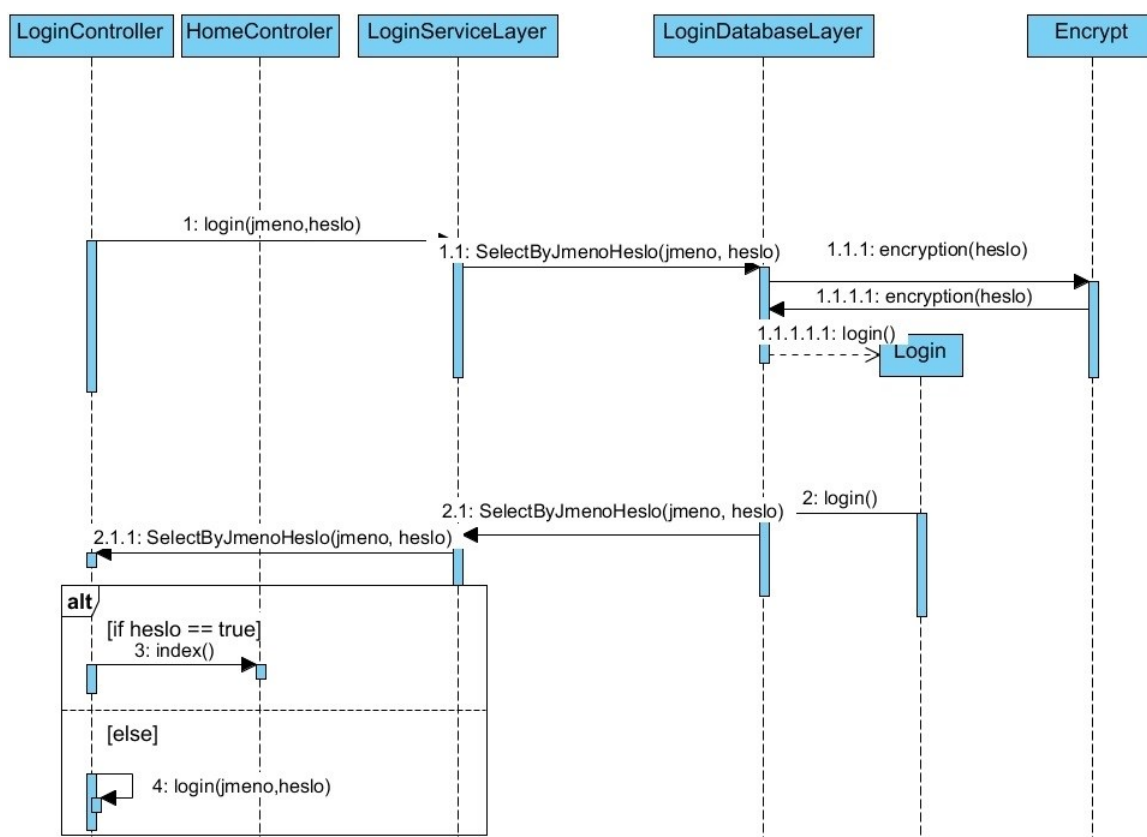
Obrázek č. 15 Diagram tříd

6 Dynamický náhled na systém

Sekvenční diagram se řadí do skupiny diagramů interakcí. Zachycuje časově uspořádanou posloupnost zasílání mezi objekty. Prvky sekvenčního diagramu jsou klasifikátory (neboli také účastníci), jejich čáry života a zprávy, které mezi sebou posílají.

Sekvenční diagram nejčastěji znázorňuje spolupráci několika objektů v rámci jednoho případu užití.

V dynamickém náhledu na systém vidíme, jak spolu komunikují jednotlivé vrstvy.



Obrázek č. 16 Sekvenční diagram přihlášení

7 Prvky architektury

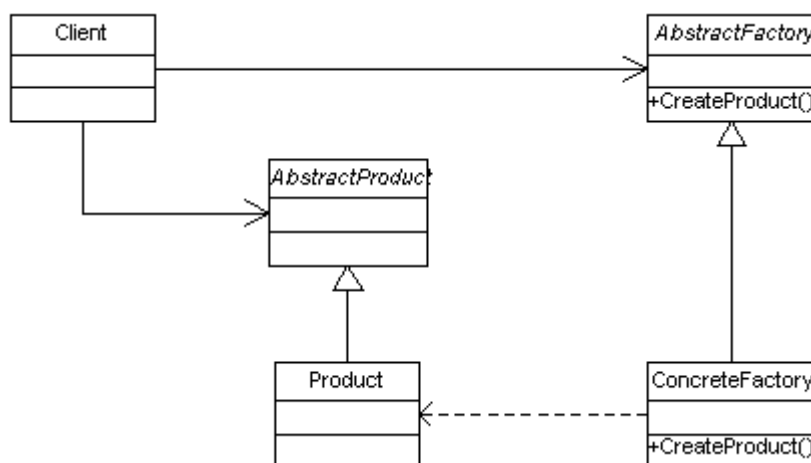
Návrhové vzory představují obecné řešení problému, které se využívá při návrhu softwaru. Jsou to ověřené programovací styly nebo postupy.

Použití návrhových vzorů je užitečné zejména proto, že jimi můžeme popsat a pochopit systém mnohem rychleji. Návrhové vzory jsou také hojně využívány pro psaní frameworků.

1. Vzor továrna:

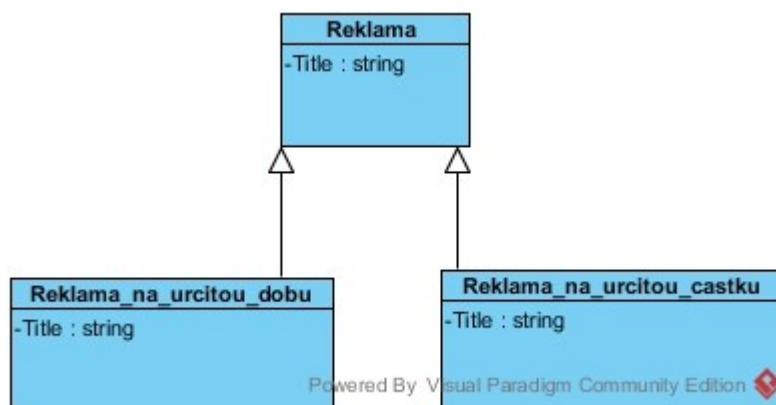
Návrhový vzor *abstract factory* (*abstraktní továrna*) je takzvaně tvořivý objektový vzor. Z tohoto zařazení vyplývá, že je určen pro vytváření instancí a že pro změnu procesu tvorby se používá změna instancí továrny.

Používá se v situacích, kdy je z nějakých dobrých důvodů vhodné delegovat vytváření nové instance třídy (tzv. produktu) na dedikovanou třídu (tzv. tovární třídu). Důvody mohou být různé - například je logika pro volbu konkrétní vytvářené třídy složitá, opakuje se na více místech, nebo ji chce továrna zkrátka "skrýt" a odstínit tak klienta od nepodstatných detailů. Tovární třída obsahuje jednu nebo více továrních metod.



Obrázek č. 17 Obecná abstraktní továrna

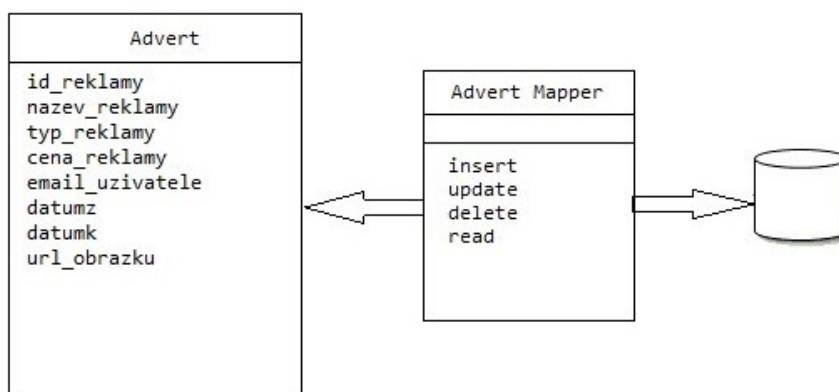
V mém případě používám abstraktní továrnu, při zadávání reklamy. Podle typu reklamy určuji, jestli se má použít buď konkrétní továrna *Reklama na určitou dobu*, nebo *Reklama na určitou částku*.



Obrázek č. 18 Konkrétní abstraktní továrna

2. Vzor data mapper:

Při použití tohoto vzoru neobsahuje doménový objekt žádné CRUD operace a o vytváření, úpravu a mazání doménových objektů z databáze se stará oddělený (mapovací) objekt. Doménový objekt je tedy zcela nezávislý na databázi. Zatímco mapovací objekt má přístup jak k doménovému objektu, tak k databázovému systému. Výhodou tohoto vzoru je právě nezávislost doménového objektu na datovém modelu, kdy je veškerá zodpovědnost za persistenci přesunuta na mapovací objekt.



Obrázek č. 19 Data mapper

8 Implementace

Pro implementaci systému jsem si vybral ASP. NET (12) (13).

ASP. NET je webový framework, stručně řečeno se jedná o sadu knihoven, které umožňují tvorbu webových aplikací v jazyce C#. Knihovny obsahují hotová řešení mnoha základních problémů, které ve webových technologiích vyvstávají. Jako např. bezpečnost, autentifikace uživatele, práce s databází, správa formulářů a podobně.

ASP. NET je založen na architektuře klient-server. Výstupem ASP. NET je tedy HTML stránka. Technologie ASP. NET tedy běží na straně serveru.

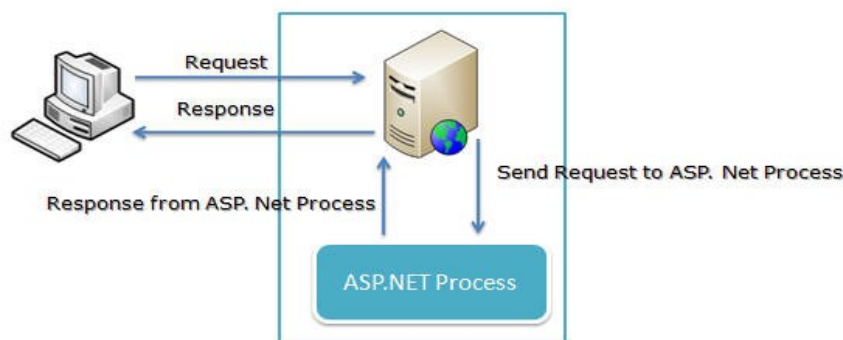
Tuto technologii můžeme ještě rozdělit na WebForms a framework MVC. V mém systému používám framework MVC.

Výhody ASP. NET:

1. ASP. NET může být použit pro rychlý vývoj aplikací (RAD), jelikož poskytuje nástroje, které umožňují rychle vytvářet aplikace, aniž bychom museli znát vnitřní mechanismus procedur.
2. Objektově orientované programování, které slouží pro design stránek a ovládacích prvků v ASP. NET umožňuje vkládat méně vstupních prvků, ale zase vede k více výstupním prvkům.
3. Platforma ASP. NET je stabilní, což zaručuje kvalitní implementaci dědění tříd.

Nevýhody ASP. NET:

1. Jednou z hlavních nevýhod ASP. NET je, že máme velmi malou kontrolu nad HTML, které je generováno nástrojem RAD (Rapid Application Development).



Obrázek č. 20 Technologie ASP. NET

Pro práci s databází jsem zvolil technologii ADO. NET, protože v rámci ASP. NET poskytuje jednoduchou a srozumitelnou komunikaci s databází.

ADO. NET je soubor typů (tříd, rozhraní, výčtů...), které můžeme použít pro přístup k datům v různých datových zdrojích z aplikací pro .NET framework. Technologie ADO. NET je spojena s formátem XML. Typy spojené s komponentou ADO. NET se nacházejí ve jmenných prostorech rozšiřující jmenný prostor System. Data.

Pro responzivní design stránek jsem použil ověřený framework Bootstrap.

Je to front-end framework, který slouží k responzivnímu designu webových stránek. Využívá HTML, CSS a Javascript. Používá tzv. mobile-first přístup, kdy se začínáme od nejmenších obrazovek (mobil) až po větší (desktop).

Všechny standartní výjimky v mém systému jsou řešeny v bloku try a catch.

Všechny třídy a funkce jsou pojmenovány konvencí CamelCase, neboli každá třída a funkce začíná velkým písmenem.

Název reklamy: Reklama12
Typ reklamy: Reklama na určitou dobu
Cena reklamy: 755
Začátek reklamy: 01.04.2016
Konec reklamy: 29.09.2016
obrazek

Reklama



Nedávné příspěvky

- [Příspěvek 1](#)
- [Příspěvek 2](#)
- [Příspěvek 3](#)

Nedávné komentáře

Obrázek č. 21 Ukázka běžící reklamy na blogu



Přihlášen uživatel [admin](#)

Přidání nové reklamy do systému

Název reklamy

Typ reklamy

Cena reklamy

Email uživatele

Obrázek č. 22 Formulář pro přidání reklamy do systému

9 SEO optimalizace

SEO (Search Engine Optimization) neboli optimalizace pro vyhledávače je způsob vytváření a upravování webových stránek takovým způsobem, aby jejich forma a obsah byly vhodné pro automatizované zpracování v internetových vyhledávačích. Cílem je získat co nejlepší pozici ve vyhledávačích, jako je Google či Seznam, po zadání určitých klíčových slov.

Základem SEO je soubor pravidel, jehož dodržování zajistí stránkám dobré umístění ve výsledcích vyhledávání.

Mezi základní pilíře kvalitního SEO patří:

- Výběr klíčových slov
- Struktura webu optimalizována pro vyhledávače
- Budování zpětných odkazů

Klíčová slova:

Správný výběr klíčových slov může zásadně ovlivnit úspěšnost optimalizace webu pro vyhledávače. Proto bychom měli dodržet několik následujících kroků:

- Začneme analýzou toho, co nabízíme. Sepíšeme si seznam produktů a služeb, které jsou důležité.
- Využíváme varianty slov. Vezmeme si náš seznam a zamyslíme se nad synonymy slov.
- Měli bychom optimalizovat na slovní spojení. Klíčová slova můžeme doplnit o přídavná jména
- Je nutno vložit nejen logo firmy, ale také název firmy.

Umístění klíčových slov:

- Do názvu stránky
- Do titulku stránky a do popisu
- Do podnadpisů

- Do samotného textu

Tvorba zpětných odkazů:

Zpětné odkazy se mohou získat např.:

- Zařazením stránek do katalogů
- Výměnou odkazů

Zařazení stránek do katalogů

Je to nejjednodušší cesta jak získat spousty odkazů.

Postup je u většiny katalogů stejný:

- Nejdříve se musíme proklikat, až do sekce kam chceme zařadit náš odkaz
- Přidáme odkaz
- Napíšeme titulek, popis stránek, adresu, email a odsouhlasíme podmínky přidání odkazu
- Za několik dní se administrátoři podívají na popis, a pokud odpovídá stránkám tak ho zařadí do katalogu
- V opačném případě pokud neodpovídá popis stránek dané sekci tak odkaz do katalogu nezařadí

Výměnou odkazů

Máme zájem o odkaz na nějakém jiném webu, nejprve musíme odkázat na web, na kterém chceme odkaz, a pak jen stačí kontaktovat majitele webu a pokud s výměnou souhlasí, přidá i náš odkaz.

Měli bychom vyměňovat odkazy s tématicky stejným webem.

10 Závěr

Zadání nebylo splněno úplně ve všech bodech, jelikož plný redakční systém by byl na implementaci příliš komplexní. Proto jsem se v mém systému, který je odlehčenou verzí Google AdWords zaměřil zejména na zadávání reklamy a její následné zobrazování na jednotlivých webových stránkách. Toto téma bakalářské práce jsem si zvolil, neboť jsem chtěl zjistit a vyzkoušet si jak fungují známé reklamní systémy a jak náročná je implementace podobného systému. Práci nejprve vedl Ing. Roman Meca, ale pak práci převzal Ing. Jan Plucar.

Nejprve jsem provedl analýzu a rešerši stávajících reklamních systémů. Z nichž je nejznámější již zmíněný Google AdWords, jenž působí po celém světě, dále pak českým zástupcem je systém s názvem Sklik. Největší výhodou Google AdWords je jeho přehledné uživatelské prostředí a následná správa reklam. Jeho další výhodou je určitě většinové zastoupení na internetu, jímž určuje ceny reklam pro ostatní reklamní systémy. Systém Sklik má tu výhodu, že je zaměřen na ryze české prostředí internetu, tím pádem se výrazně zmenšuje okruh potencionálních zákazníků. Jeho nevýhodou je zcela jistě nekonkurenceschopnost v zahraničí, kde převládá Google AdWords.

Dále jsem pokračoval analýzou mého systému, kde jsem si určil požadavky, které je nutno klást na systém. Navrhl jsem korektní architektonické řešení tzn. že jsem pro svůj systém zvolil třívrstvou architekturu. Seznámil jsem se s novými technologiemi, kterými jsou například framework ASP. NET MVC, technologie ADO. NET a framework Bootstrap pro responzivní design webových stránek.

Systém zdaleka není plný redakční systém, a proto jsem se po konzultaci s vedoucím práce rozhodl zaměřit na části kde je reklama zadávána do systému a následně podle kategorie, kterou si vybere zákazník je za menší poplatek zákazníkovi vytvořen blog se stejnou kategorií jakou má reklama. Tento systém, který implementuji v bakalářské práci bych pak dále chtěl rozšířit v rámci diplomové práce. Screeny ze systému lze vidět v kapitole implementace.

11 Seznam použité literatury

1. **Irvine, Mark.** Average Cost per Click by Country: Where in the World Are the Highest CPCs? *Wordstream*. [Online] 6. 7 2015.
<http://www.wordstream.com/blog/ws/2015/07/06/average-cost-per-click>.
2. **Shewan, Dan.** How Much Does Google AdWords Cost? *Wordstream*. [Online] 21. 5 2015.
<http://www.wordstream.com/blog/ws/2015/05/21/how-much-does-adwords-cost>.
3. **Větrovská, Petra .** Návrh klíčových slov v Skliku. *Zaklik*. [Online] 26. 1 2015.
<http://www.zaklik.cz/klicova-slova/navrh-klicovych-slov-v-skliku/>.
4. **Leffingwell, Dean.** *Agile software requirements: lean requirements practices for teams, programs, and the enterprise*. Upper Saddle River, NJ : Addison-Wesley, 2011. ISBN 0321635841.
5. **Kaufmann, Morgan.** *Agile Systems Engineering*. Waltham : Elsevier, 2015. ISBN 0128021209.
6. **Rosenberg, Doug a Stephens, Matt.** *Use case driven object modeling with UML: theory and practice*. 2. Berkeley, CA : Apress, 2013. ISBN 9781590597743.
7. **Blokdijk, Gerard.** *Unified modeling language - Simple Steps to Win, Insights and Opportunities for Maxing Out Success*. 2015. ISBN 9781489102300.
8. **Tsui, Frank F, Karam, Orlando a Bernal, Barbara.** *Essentials of software engineering*. 3. Burlington, MA : Jones & Bartlett Learning, 2014. ISBN 1449691994.
9. **Roudenský, Petr a Havlíčková, Anna.** *Řízení kvality softwaru: průvodce testováním*. Brno : Computer Press, 2013. ISBN 9788025138168.
10. **Bell, Michael.** *Incremental software architecture: a method for saving failing IT implementations*. New Jersey : Wiley, 2016. ISBN 9781119117643.
11. **Esposito, Dino a Saltarello, Andrea.** *Microsoft.NET: architecting applications for the enterprise*. 2. Washington : Microsoft Press, 2014. ISBN 0735685355.
12. **Esposito, Dino.** *Programming Microsoft ASP.NET MVC*. 3. Sebastopol, California : O'Reilly Media, Inc., 2014. ISBN 0735680949.
13. **Esposito, Dino.** *Modern web development*. Redmond, WA : Microsoft Press, 2015. ISBN 9781509300532.

Přílohy na CD/DVD

Obsah CD/DVD:

- SmartAdvertising.GuiMvc – složka systému
- software requirements set.docx – všechny softwarové požadavky
- obrázky, které se nevešly do textu bakalářské práce